UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/824,809	04/15/2004	Cliff Daniel Cyphers	AUS920040045US1	6688

45371 7590 08/24/2007
IBM CORPORATION (RUS)
c/o Rudolf O Siegesmund Gordon & Rees, LLp
2100 Ross Avenue
Suite 2600
DALLAS, TX 75201

EXAMINER	
WANG, JUE S	

ART UNIT	PAPER NUMBER
2193	

MAIL DATE	DELIVERY MODE
08/24/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	**Application No.** 10/824,809	**Applicant(s)** CYPHERS, CLIFF DANIEL	
	Examiner Jue S. Wang	**Art Unit** 2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after t he mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _15 April 2004_.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-20_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-20_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☒ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _15 April 2004_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _15 April 2004_.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-20 have been examined.

Specification

2. The specification is objected to because of the following informalities:

3. The use of the trademark Java has been noted in this application. It should be capitalized wherever it appears and be accompanied by the generic terminology.

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as trademarks.

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 1-20 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

A. The following lacks antecedent basis in the claims:

i. Claim 8, “The program product claim 1” in line 1. Claim 1 is a method claim and does not recite a program product. Because of the recitation of “a program product” in claim 7, it is believed that claim 8 was intended to depend on claim 7 and it is treated as such for compact prosecution of the claims.

B. The following claim language is not clear and indefinite:

i. As per claim 1, line 2, claim 2, lines 2, 7, claim 7, line 2, claim 8, lines 5, 10, claim 13, line 8, claim 15, lines 3, 8, and claim 20, lines 2, 6, 12, the phrase "JAR file structure" is used. This limitation is not clearly understood because it is not clear what is considered the JAR file structure (i.e., the directory structure within the unpacked JAR file, or the ordering and grouping of every file within the unpacked JAR file?).

ii. As per claim 2, line 3, claim 8, line 6, claim 15, line 4, claim 20, line 7, the phrase "extracting the JAR file content" is used. This limitation is not clearly understood because it is not clear what JAR file content is extracted (i.e., is every file within the JAR file extracted effectively unpacking the JAR file, or are only specific files extracted, and if so, which files within the JAR file are extracted?).

iii. As per claim 3, line 2, claim 10, line 2, claim 17, line 2, and claim 20, line 9, the phrase "user specification of a file" is used. This limitation is not clearly understood because it is not clear whether the file is one of the files located in the JAR file, or the file is a JAR file.

iv. As per claim 3, lines 2, 3, claim 10, lines 2, 3, claim 17, lines 2, 3, and claim 20, lines 9, 10, the term "field" is used. This limitation is not clearly understood because it is not clear what is considered a field (i.e., a tag for indicating a particular type of data, or a specific region within a file?).

C. Claims 1, 7, 13, and 20 contain the trademark/trade name Java. Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. See *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name Java is used to identify/describe a type of archive file and, accordingly, the identification/description is indefinite.

Appropriate corrections are required.

Any claim not specifically addressed, above, is being rejected as incorporating the deficiencies of a claim upon which it depends.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

7. Claims 7 and 13-20 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

8. Claim 7 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. In claim 7, the "computer program product" recited is a program which is software, per se.

9. Claim 13 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. In claim 13, the "Graphical User Interface" recited would reasonably be interpreted by one of ordinary skill in the art as software, per se, since the file specification section, the field specification section, the old value specification sections and the new value specification section would reasonably be interpreted by one of ordinary skill in the art as software, per se. As such, it is believed that the Graphical User Interface is reasonably interpreted as functional descriptive material, per se, failing to be tangibly embodied or include any recited hardware.

10. Claims 14-19 fail to resolve the deficiencies of claim 13. Claims 14-19 disclose additional features of Graphical User Interface. The limitations recited in claims 14-19 do not invalidate the reasonable interpretation of the system as software, per se, by one of ordinary skill in the art.

11. Claim 20 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. In claim 13, an "apparatus" is recited; however, it appears that the apparatus would reasonably be interpreted by one of ordinary skill in the art as software, per se, since the means for creating, recording, extracting, accepting, searching, replacing, archiving,

and determining would reasonably be interpreted by one of ordinary skill in the art as software, per se. As such, it is believed that the apparatus is reasonably interpreted as functional descriptive material, per se, failing to be tangibly embodied or include any recited hardware.

Claim Rejections - 35 USC § 102

12. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

13. Claims 1 and 7 are rejected under 35 U.S.C. 102(e) as being anticipated by Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy).

14. As per claim 1, Narayanaswamy teaches the invention as claimed including a method for updating a Java Archive (JAR) file content wherein the data within the JAR file is changed and wherein the JAR file structure remains the same before and after the data is changed (see Fig 7, Fig 8, column 2, lines 3-6, 38-41, column 5, lines 23-35, column 5, line 57 – column 6, line 3, column 6, lines 36-58, column 7, lines 59-61, column 8, lines 12-15, and column 16, 38-63; EN: updating the descriptor files in the JAR files of an EAR file is considered as updating a JAR file content, and the JAR file structure remains the same before and after the data is changed because only the data within the deployment descriptor is changed as a result of the update).

15. As per claim 7, this is the program product claim of claim 1. Therefore, it is rejected using the same reasons as claim 1.

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

17. Claims 2 and 8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy).

18. As per claim 2, Narayanaswarmy further teaches

recording the JAR file structure (see Fig 7, column 6, lines 36-39, and column 16, lines 38-44; EN: the JAR file structure must be recorded since EJB display tree containing the JAR file structure is displayed);

extracting the JAR file content (see Fig 7, Fig 8, column 6, lines 36-48, and column 8, lines 39-42, extracting the deployment descriptor in a JAR file);

accepting specification of an old value and a new value (see column 8, lines 37-53, column 11, lines 7-41, "A string that may need to be replaced" is the old value and "the replacement string" is the new value);

searching the JAR file content for the specified old value (see column 8, lines 37-49, "The checkForStart and CheckForEnd methods of the Replacer interface are called to locate the beginning and end of the string that may need to be replaced");

replacing the specified old value with the specified new value (see column 8, lines 37-55, column 11, lines 7-52, and column 14, lines 14-18);

archiving the JAR file content according to the recorded JAR file structure (see column 7, lines 58-61 and column 8, lines 12-16; EN: the EAR file is repackaged with the modified deployment descriptors where the JAR file content must be archived according to the original JAR file construct since only the deployment descriptors are modified).

Narayanaswarmy does not explicitly teach that the old value and the new value are specified by the user. However, it would have been obvious to one of ordinary skill in the art at the time of the invention that the old value and the new value can be specified by the user since the purpose of the invention was to provide the user with a series of input tools or panels for specifying deployment variables and customizing the deployment as needed (see abstract, lines 6-8, column 5, lines 23-35, and column 8, lines 37-47).

19. As per claim 8, this is the program product claim of claim 2. Therefore, it is rejected using the same reasons as claim 2.

20. Claims 3, 10, and 12-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy), in view of Chan et al. (US 2002/0129053 A1, hereinafter Chan).

21. As per claim 3, Narayanaswarmy does not teach accepting a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user.

Chan teaches a search and replace feature for excel sheets that accepts a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user (see Fig 13, abstract, lines 1-4, [0042], and [0043]; EN: the "Within" option (Fig 13, item 308) specifies a worksheet to search in which is a file, and the "Look in" option (Fig 13, item 312) specifies the field).

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy to accept a user specification of a file and a field and searching for the old value only in the file and the field specified by the user as taught by Chan because it provides the user greater control over where the search and replace operation should occur so that when a user is searching for a particular term, the user can limit the multiple instances of the term to reach the particular instanced desired (see [0003] of Chan).

22. As per claim 10, this is the program product claim of claim 3. Therefore, it is rejected using the same reasons as claim 3.

23. As per claim 12, Narayanaswarmy does not teach determining if the user desires to update another value; and responsive to the determination that the user desires to update another value; accepting a user specification of another old value and another new value.

Chan teaches determining if the user desires to update another value; and responsive to the determination that the user desires to update another value; accepting a user specification of another old value and another new value (see Fig 13, abstract, lines 1-4, [0042], and [0043]; EN: while Chan does not explicitly teach as such, it is well known in the art that entering new values in the "Find What" and "Replace With" fields and pressing the "Replace" button will indicate that the user desired to update another value and the search and replace function will accept the old value and the new value).

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy to determine if the user desires to update another value, and responsive to the determination that the user desires to update another value, accept a user specification of another old value and another new value as taught by Chan because it allows the user to update more than one value in a single session.

24. As per claim 13, Narayanaswarmy teaches the invention as claimed, including a Graphical User Interface (GUI) that allows a user to update a value within a Java Archive (JAR) file (see Fig 9, abstract, lines 6-8, column 5, lines 23-35, and column 8, lines 37-47), the GUI

accepts a specification of an old value and a new value (see column 8, lines 37-53, column 11, lines 7-41, "A string that may need to be replaced" is the old value and "the replacement string" is the new value);

replaces the specified old value with the specified new value (see column 8, lines 37-55, column 11, lines 7-52, and column 14, lines 14-18);

wherein the JAR file structure remains the same before and after the data is changed (see Fig 7, Fig 8, column 2, lines 3-6, 38-41, column 5, lines 23-35, column 5, line 57 – column 6, line 3, column 6, lines 36-58, column 7, lines 59-61, column 8, lines 12-15, and column 16, 38-

63; EN: updating the descriptor files in the JAR files of an EAR file is considered as updating a JAR file content, and the JAR file structure remains the same before and after the data is changed because only the data within the deployment descriptor is changed as a result of the update)

Narayanaswarmy does not explicitly teach that the GUI has a file specification section containing a file located in the JAR file, a field specification section containing a field located in the file, an old value specification section containing an old value located in the field, and a new value specification section.

Chan teaches a search and replace feature for excel sheets with a file specification section containing a file located in a workbook, a field section containing a field located in the file, an old value specification section containing an old value located in the filed, a new value specification section containing the new value (see Fig 13, items 207, 1202, 308, 312, abstract, lines 1-4, [0042], and [0043]; EN: the "Within" option (Fig 13, item 308) specifies a worksheet (considered as a file) within a workbook to search in, and the "Look in" option (Fig 13, item 312) specifies the field).

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy such that the GUI has a file specification section containing a file located in the JAR file, a field specification section containing a field located in the file, an old value specification section containing an old value located in the field, and a new value specification section as taught by Chan because it is well known in the art that a common layout in a GUI for search and replace will have sections to enter the desired information. Furthermore, it would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy to accept a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user as taught by Chan because it provides the user greater control over where the search and replace operation should occur so that when a user is searching for a particular term, the user can limit the multiple instances of the term to reach the particular instanced desired (see [0003] of Chan).

25. As per claim 14, Narayanaswarmy does not teach a values updated section; and wherein the values updated section records the history of the replacement of the old values by the new values.

Chan does not explicitly teach a values updated section; and wherein the values updated section records the history of the replacement of the old values by the new values. However, Chan teaches a dropdown button that lists previous searches entered into the find what field (see Fig 13, item 214, [0035], [0073]) and the replace with field has a similar dropdown button (see Fig 13). Therefore, while Chan does not explicitly include a values updated section in its GUI, the combination of the dropdown buttons for the find what field and the replace with field performs the functionality of the values updated section.

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy such that the GUI has a values updated section; and wherein the values updated section records the history of the replacement of the old values by the new values as taught by Chan because a user may quickly select a previous search term and replace term and perform search and replace for those term again (see [0035] of Chan).

26. As per claim 15, Narayanaswarmy teaches the indication by a user to replace the old value with the new value causes a processor to perform steps comprising:

recording the JAR file structure (see Fig 7, column 6, lines 36-39, and column 16, lines 38-44);

extracting the JAR file content (see Fig 7, Fig 8, column 6, lines 36-48, and column 8, lines 43);

accepting a specification of an old value and a new value (see column 8, lines 37-53, column 11, lines 7-41);

searching the JAR file content for the specified old value (see column 8, lines 37-49;

replacing the specified old value with the specified new value (see column 8, lines 37-55, column 11, lines 7-52, and column 14, lines 14-18); and

archiving the JAR file content according to the recorded JAR file structure (see column 7, lines 58-61 and column 8, lines 12-16).

Narayanaswarmy does not explicitly teach that the old value and the new value are specified by the user. However, it would have been obvious to one of ordinary skill in the art at the time of the invention that the old value and the new value can be specified by the user since the purpose of the invention was to provide the user with a series of input tools or panels for specifying deployment variables and customizing the deployment as needed (see abstract, lines 6-8, column 5, lines 23-35, and column 8, lines 37-47).

27. Claims 4-6, and 16-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy), in view of Chan et al.

(US 2002/0129053 A1, hereinafter Chan), as applied to claims 3 and 15 above, further in view of Kronenberg et al. (US 6,405,265 B1, hereinafter Kronenberg).

28. As per claim 4, Narayanaswarmy and Chan do not teach that the JAR file content are extracted into a temporary directory.

Kronenberg teaches a method of accessing and updating archive files including extracting archive file content into a temporary directory (see column 2, line 61 – column 3, line 43, column 5, lines 1-7, 29-32; the virtual folder is the temporary directory).

It would have been obvious of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy such that the JAR file content are extracted into a temporary directory as taught by Kronenberg because the temporary directory presents a common interface to the archive file so that the contents of an archive are made available throughout the operating system (see column 5, lines 26-33 of Kronenberg).

29. As per claim 5, Narayanaswarmy and Kronenberg do not teach determining if the user desires to update another value; and responsive to the determination that the user desires to update another value; accepting a user specification of another old value and another new value.

Chan teaches determining if the user desires to update another value; and responsive to the determination that the user desires to update another value; accepting a user specification of another old value and another new value (see Fig 13, abstract, lines 1-4, [0042], and [0043]; EN: while Chan does not explicitly teach as such, it is well known in the art that entering new values in the "Find What" and "Replace With" fields and pressing the "Replace" button will indicate that the user desired to update another value and the search and replace function will accept the old value and the new value).

30. As per claim 6, Narayanaswarmy and Chan do not teach creating the temporary directory and copying the JAR file into the temporary directory.

Kronenberg teaches creating a temporary file (see column 2, line 61 – column 3, line 6). Kronenberg only teaches that the temporary directory represents the archive file (see column 2, lines 61-63) and does not specifically teach copying the archive file into the temporary directory. However, it would have been obvious to one of ordinary skill in the art at the time of the invention to copy the archive file into the temporary directory since extracting files from a copy of the archive file already in the temporary directory will be faster than extracting files from the original copy of the archive file in the storage device.

31. As per claim 16, the limitations recited in this claim are substantially similar to claim 6. Therefore, it is rejected using the same reasons as claim 6.

32. As per claim 17, Narayanaswarmy does not teach accepting a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user.

Chan teaches a search and replace feature excel sheets that accepts a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user (see Fig 13, abstract, lines 1-4, [0042], and [0043]; EN: the "Within" option (Fig 13, item

308) specifies a worksheet to search in which is a file, and the "Look in" option (Fig 13, item 312) specifies the field).

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy such to accept a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user as taught by Chan because it provides the user finer control over where the search and replace operation should occur so that when a user is searching for a particular term, the user can limit the multiple instances of the term to reach the particular instanced desired (see [0003] of Chan).

33. As per claim 18, the limitations recited in this claim are substantially similar to claim 3. Therefore, it is rejected using the same reasons as claim 3.

34. As per claim 19, the limitations recited in this claim are substantially similar 5. Therefore, it is rejected using the same reasons as claim 5.

35. Claims 9 and 11 are rejected under 35 U.S.C. 103(a) as being unpatentable over Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy), in view of Kronenberg et al. (US 6,405,265 B1, hereinafter Kronenberg).

36. As per claim 9, Narayanaswarmy does not teach creating a temporary directory; and copying the JAR file into the temporary directory.

Kronenberg teaches creating a temporary file (see column 2, line 61 – column 3, line 6). Kronenberg only teaches that the temporary directory represents the archive file (see column 2, lines 61-63) and does not specifically teach copying the archive file into the temporary directory. However, it would have been obvious to one of ordinary skill in the art at the time of the invention to copy the archive file into the temporary directory since extracting files from a copy of the archive file already in the temporary directory will be faster than extracting files from the original copy of the archive file in the storage device.

It would have been obvious of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy to create a temporary directory and copy the JAR file into the temporary directory as taught by Kronenberg because the temporary directory presents a common interface to the archive file so that the contents of an archive are made available throughout the operating system (see column 5, lines 26-33 of Kronenberg).

37. As per claim 11, Narayanaswarmy does not teach that the JAR file content are extracted into the temporary directory.

Kronenberg teaches a method of accessing and updating archive files including extracting archive file content into a temporary directory (see column 2, line 61 – column 3, line 43, column 5, lines 1-7, 29-32; the virtual folder is the temporary directory).

It would have been obvious of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy such that the JAR file content are extracted into a temporary directory as taught by Kronenberg because the temporary directory presents a common interface to the archive file so that the contents of an archive are made available throughout the operating system (see column 5, lines 26-33 of Kronenberg).

38. Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over Narayanaswamy et al. (US 7,069,553 B2, hereinafter Narayanaswamy), in view of Chan et al. (US 2002/0129053 A1, hereinafter Chan), further in view of Kronenberg et al. (US 6,405,265 B1, hereinafter Kronenberg).

39. As per claim 20, Narayanaswarmy teaches the invention as claimed, including an apparatus for updating a Java Archive (JAR) file content wherein the data within the JAR file is changed and wherein the JAR file structure remains the same before and after the data is changed (see Fig 7, Fig 8, column 2, lines 3-6, 38-41, column 5, lines 23-35, column 5, line 57 – column 6, line 3, column 6, lines 36-58, column 7, lines 59-61, column 8, lines 12-15, and column 16, 38-63; EN: updating the descriptor files in the JAR files of an EAR file is considered as updating a JAR file content, and the JAR file structure remains the same before and after the data is changed because only the data within the deployment descriptor is changed as a result of the update), the apparatus comprising:

means for recording the JAR file structure (see Fig 7, column 6, lines 36-39, and column 16, lines 38-44; EN: the JAR file structure must be recorded since EJB display tree containing the JAR file structure is displayed);

means for extracting the JAR file content (see Fig 7, column 6, lines 36-39, and column 16, lines 38-44);

accepting specification of an old value and a new value (see column 8, lines 37-53, column 11, lines 7-41;

searching the JAR file content for the specified old value (see column 8, lines 37-49);

replacing the specified old value with the specified new value (see column 8, lines 37-55, column 11, lines 7-52, and column 14, lines 14-18);

archiving the JAR file content according to the recorded JAR file structure (see column 7, lines 58-61 and column 8, lines 12-16).

Narayanaswarmy does not explicitly teach that the old value and the new value are specified by the user. However, it would have been obvious to one of ordinary skill in the art at the time of the invention that the old value and the new value can be specified by the user since the purpose of the invention was to provide the user with a series of input tools or panels for specifying deployment variables and customizing the deployment as needed (see abstract, lines 6-8, column 5, lines 23-35, and column 8, lines 37-47).

Narayanaswarmy does not explicitly teach that the old value and the new value are specified by the user. However, it would have been obvious to one of ordinary skill in the art at the time of the invention that the old value and the new value can be specified by the user since the purpose of the invention was to provide the user with a series of input tools or panels for specifying deployment variables and customizing the deployment as needed (see abstract, lines 6-8, column 5, lines 23-35, and column 8, lines 37-47).

Narayanaswarmy also does not teach accepting a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user.

Chan teaches a search and replace feature for excel sheets that accepts a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user (see Fig 13, abstract, lines 1-4, [0042], and [0043]; EN: the "Within" option (Fig 13, item 308) specifies a worksheet to search in which is a file, and the "Look in" option (Fig 13, item 312) specifies the field).

It would have been obvious to one of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy to accept a user specification of a file and a field; and searching for the old value only in the file and the field specified by the user as taught by Chan because it provides the user greater control over where the search and replace operation should occur so that when a user is searching for a particular term, the user can limit the multiple instances of the term to reach the particular instanced desired (see [0003] of Chan).

Narayanaswarmy and Chan do not teach means for creating a temporary directory; means for copying the JAR file into the temporary directory; and that the JAR file content are extracted into the temporary directory.

Kronenberg teaches a method of accessing and updating archive files including creating a temporary file (see column 2, line 61 – column 3, line 6, the virtual folder is the temporary directory) and extracting archive file content into a temporary directory (see column 2, line 61 – column 3, line 43, column 5, lines 1-7, 29-32). Kronenberg does not specifically teach copying the archive file into the temporary directory. However, it would have been obvious to one of ordinary skill in the art at the time of the invention to copy the archive file into the temporary directory since extracting files from a copy of the archive file already in the temporary directory will be faster than extracting files from the original copy of the archive file in the storage device.

It would have been obvious of ordinary skill in the art at the time of the invention to have modified Narayanaswarmy as modified to include means for creating a temporary directory; means for copying the JAR file into the temporary directory; and that the JAR file content are extracted into the temporary directory as taught by Kronenberg because the temporary directory presents a common interface to the archive file so that the contents of an archive are made available throughout the operating system (see column 5, lines 26-33 of Kronenberg).

Conclusion

40. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

-
- Becker et al. (US 6,286,051 B1) is cited to teach a method and apparatus for extending a Java Archive file.
- Rodriguez et al. (US 6,427,149 B1) is cited to teach remote access of archived compressed data files.
- Schmidt et al. (US 6,535,894 B1) is cited to teach an apparatus and method for incremental updating of archive files.
- Basin et al. (US 6,879,988 B2) is cited to teach a system and method for manipulating and managing computer archive files.

41. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Jue S. Wang whose telephone number is (571) 270-1655. The examiner can normally be reached on M-Th 7:30 am - 5:00pm (EST).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Jue Wang
Examiner
Art Unit 2193

MENG-AI T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100